CLAIMS

1. An object-oriented programming apparatus for interconnecting a plurality of objects each having data and operations, said object-oriented programming apparatus comprising:

instruction coupling means for permitting a transfer of messages between a first object having an output instruction bus portion for performing a processing for an issue of messages directed to another object and a second object having an input instruction bus portion responsive to messages issued by another object and directed to self object for activating a method of self object associated with the received message, by means of providing such a correspondence that the message of the first object is associated with the method of the second object;

data element list generating means for generating a data element list, in which pointers to data storage areas for storing data are arranged, of an object;

pointer element list generating means for generating a pointer element list, in which pointers to pointer storage areas for storing pointers to data are arranged, of an object; and

data coupling means for permitting a transfer of data between a third object having the data element list and a fourth object having the pointer element list, by means of writing the pointers arranged in the data element list of the third object into the pointer storage areas indicated by the pointers arranged in the pointer element list of the fourth object.

2.   An object-oriented programming apparatus according to claim 1, wherein said instruction coupling means generates a method element list in which arranged are method elements including a method ID for specifying a method of another object associated with a message of self object, and a pointer to another object in which the method specified by the method ID is executed.

3. An object-oriented programming apparatus for interconnecting a plurality of objects each having data and operations, said object-oriented programming apparatus comprising:

instruction coupling means for permitting a transfer of messages between a first object having an output instruction bus portion for performing a processing for an issue of messages directed to another object and a second

197

object having an input instruction bus portion responsive to messages issued by another object and directed to self object for activating a method of self object associated with the received message, by means of providing such a correspondence that the message of the first object is associated with the method of the second object; and

an input instruction tag table generating means for generating an input instruction tag table indicating an association of messages of another object with methods of self object, for each other object, on the output instruction bus portion of self object.

4.   An object-oriented programming apparatus according to claim 3, wherein said instruction coupling means generates a method element list in which arranged are method elements including a method ID for specifying a method of another object associated with a message of self object, and a pointer to another object in which the method specified by the method ID is executed, and

wherein said input instruction tag table generating means generates the input instruction tag table and adds the input instruction tag table to the method elements including the pointer to another object associated with the input instruction tag table.

5.   An object-oriented programming apparatus for interconnecting a plurality of objects each having data and operations, said object-oriented programming apparatus comprising:

instruction coupling means for permitting a transfer of messages between a first object having an output instruction bus portion for performing a processing for an issue of messages directed to another object and a second object having an input instruction bus portion responsive to messages issued by another object and directed to self object for activating a method of self object associated with the received message, by means of providing such a correspondence that the message of the first object is associated with the method of the second object; and

an output instruction tag table generating means for generating an output instruction tag table indicating an association of methods of another object with messages of self object, for each other object, on the output instruction bus portion of self object.

6.   An object-oriented programming apparatus according to claim 5, wherein said instruction coupling means generates a method element list in which arranged are method

199

elements including a method ID for specifying a method of another object associated with a message of self object, and a pointer to another object in which the method specified by the method ID is executed, and

wherein said output instruction tag table generating means generates the output instruction tag table and adds the output instruction tag table to the method elements including the pointer to another object associated with the output instruction tag table.

7. An object-oriented programming apparatus for interconnecting a plurality of objects each having data and operations, said object-oriented programming apparatus comprising:

instruction coupling means for permitting a transfer of messages between a first object having an output instruction bus portion for performing a processing for an issue of messages directed to another object and a second object having an input instruction bus portion responsive to messages issued by another object and directed to self object for activating a method of self object associated with the received message, by means of providing such a correspondence that the message of the first object is associated with the method of the second object; and

200

an input data tag table generating means for

generating an input data tag table indicating an association

of a data element list ID for identifying a data element

list in which pointers to data storage areas for storing data

are arranged with a pointer element list ID for identifying

a pointer element list in which pointers to data storage

areas for storing pointer to data are arranged, for each

other object, on the output instruction bus portion of self

object.

8.   An object-oriented programming apparatus

according to claim 7, wherein said instruction coupling means

generates a method element list in which arranged are method

elements including a method ID for specifying a method of

another object associated with a message of self object, and

a pointer to another object in which the method specified by

the method ID is executed, and

wherein said input data tag table generating means

generates the input data tag table and adds the input data

tag table to the method elements including the pointer to

another object associated with the input data tag table.

9.   An object-oriented programming apparatus for

interconnecting a plurality of objects each having data and

operations, said object-oriented programming apparatus comprising:

instruction coupling means for permitting a transfer of messages between a first object having an output instruction bus portion for performing a processing for an issue of messages directed to another object and a second object having an input instruction bus portion responsive to messages issued by another object and directed to self object for activating a method of self object associated with the received message, by means of providing such a correspondence that the message of the first object is associated with the method of the second object; and

an output data tag table generating means for generating an output data tag table indicating an association of a pointer element list ID for identifying a pointer element list in which pointers to pointer storage areas for storing pointers to data are arranged with a data element list ID for identifying a data element list in which pointers to data storage areas for storing data are arranged, for each other object, on the output instruction bus portion of self object.

10. An object-oriented programming apparatus according to claim 9, wherein said instruction coupling means generates a method element list in which arranged are method elements including a method ID for specifying a method of another object associated with a message of self object, and a pointer to another object in which the method specified by the method ID is executed, and

wherein said output data tag table generating means generates the output data tag table and adds the output data tag table to the method elements including the pointer to another object associated with the output data tag table.

11. An object-oriented program storage medium for storing a plurality of objects each having data and operations, said object-oriented program storage medium storing an object coupling program comprising:

instruction coupling means for permitting a transfer of messages between a first object having an output instruction bus portion for performing a processing for an issue of messages directed to another object and a second object having an input instruction bus portion responsive to messages issued by another object and directed to self object for activating a method of self object associated with the received message, by means of providing such a correspondence

that the message of the first object is associated with the method of the second object;

data element list generating means for generating a data element list, in which pointers to data storage areas for storing data are arranged, of an object;

pointer element list generating means for generating a pointer element list, in which pointers to pointer storage areas for storing pointers to data are arranged, of an object; and

data coupling means for permitting a transfer of data between a third object having the data element list and a fourth object having the pointer element list, by means of writing the pointers arranged in the data element list of the third object into the pointer storage areas indicated by the pointers arranged in the pointer element list of the fourth object.

12. An object-oriented program storage medium according to claim 11, wherein said instruction coupling means generates a method element list in which arranged are method elements including a method ID for specifying a method of another object associated with a message of self object, and a pointer to another object in which the method specified by the method ID is executed, and

wherein the first object having the output instruction bus portion refers to, when issuing a message, a method element arranged in the method element list associated with the message, and calls the second object in which a pointer is stored in the method element, giving the method ID stored in the method element as an argument.

13.   An object-oriented program storage medium for storing a plurality of objects each having data and operations, said object-oriented program storage medium storing an object coupling program comprising:

instruction coupling means for permitting a transfer of messages between a first object having an output instruction bus portion for performing a processing for an issue of messages directed to another object and a second object having an input instruction bus portion responsive to messages issued by another object and directed to self object for activating a method of self object associated with the received message, by means of providing such a correspondence that the message of the first object is associated with the method of the second object; and

an input instruction tag table generating means for generating an input instruction tag table indicating an association of messages of another object with methods of

205

self object, for each other object, on the output instruction
bus portion of self object.

14.   An object-oriented program storage medium
according to claim 13, wherein said instruction coupling
means generates a method element list in which arranged are
method elements including a method ID for specifying a
method of another object associated with a message of self
object, and a pointer to another object in which the method
specified by the method ID is executed, and

wherein said input instruction tag table generating
means generates the input instruction tag table and adds the
input instruction tag table to the method elements including
the pointer to another object associated with the input
instruction tag table.

15.   An object-oriented program storage medium
according to claim 14, wherein the first object having the
method element to which the input instruction tag table is
added calls, when calling the second object identified by the
method element, the second object giving as arguments the
method ID and the input instruction tag table which are
stored in the method element.

16. An object-oriented program storage medium according to claim 15, wherein the second object receives messages directed from the first object to the second object, and refers to the input instruction tag table, which is an argument of the received message, to execute the method of the first object associated with the message of the second object.

17. An object-oriented program storage medium according to claim 15, wherein the second object receives messages directed from the first object to the second object, and refers to the input instruction tag table, which is an argument of the received message, to add the method element related to the method of the first object associated with the message of the second object to the method element list of the second object associated with the message of the second object.

18. An object-oriented program storage medium according to claim 15, wherein the second object has means for producing a third object, receives messages directed from the first object to the second object, and refers to the input instruction tag table, which is an argument of the received message, to add the method element related to the

207

method of the first object associated with messages of the third object to the method element list of the third object associated with the message of the third object.

19. An object-oriented program storage medium for storing a plurality of objects each having data and operations, said object-oriented program storage medium storing an object coupling program comprising:

instruction coupling means for permitting a transfer of messages between a first object having an output instruction bus portion for performing a processing for an issue of messages directed to another object and a second object having an input instruction bus portion responsive to messages issued by another object and directed to self object for activating a method of self object associated with the received message, by means of providing such a correspondence that the message of the first object is associated with the method of the second object; and

an output instruction tag table generating means for generating an output instruction tag table indicating an association of methods of another object with messages of self object, for each other object, on the output instruction bus portion of self object.

208

20. An object-oriented program storage medium according to claim 19, wherein said instruction coupling means generates a method element list in which arranged are method elements including a method ID for specifying a method of another object associated with a message of self object, and a pointer to another object in which the method specified by the method ID is executed, and

wherein said output instruction tag table generating means generates the output instruction tag table and adds the output instruction tag table to the method elements including the pointer to another object associated with the output instruction tag table.

21. An object-oriented program storage medium according to claim 20, wherein the first object having the method element to which the output instruction tag table is added calls, when calling the second object identified by the method element, the second object giving as arguments the method ID and the output instruction tag table which are stored in the method element.

22. An object-oriented program storage medium according to claim 21, wherein the second object receives messages directed from the first object to the second object,

209

and refers to the output instruction tag table, which is an argument of the received message, to add the method element related to the method of the second object associated with the message of the first object to the method element list of the first object associated with the message of the first object.

23.   An object-oriented program storage medium according to claim 21, wherein the second object has means for producing a third object, receives messages directed from the first object to the second object, and refers to the output instruction tag table, which is an argument of the received message, to add the method element related to the method of the third object associated with messages of the first object to the method element list of the first object associated with the message of the first object.

24.   An object-oriented program storage medium for storing a plurality of objects each having data and operations, said object-oriented program storage medium storing an object coupling program comprising:

instruction coupling means for permitting a transfer of messages between a first object having an output instruction bus portion for performing a processing for an

issue of messages directed to another object and a second object having an input instruction bus portion responsive to messages issued by another object and directed to self object for activating a method of self object associated with the received message, by means of providing such a correspondence that the message of the first object is associated with the method of the second object; and

an input data tag table generating means for generating an input data tag table indicating an association of a data element list ID for identifying a data element list in which pointers to data storage areas for storing data are arranged with a pointer element list ID for identifying a pointer element list in which pointers to data storage areas for storing pointer to data are arranged, for each other object, on the output instruction bus portion of self object.

25. An object-oriented program storage medium according to claim 24, wherein said instruction coupling means generates a method element list in which arranged are method elements including a method ID for specifying a method of another object associated with a message of self object, and a pointer to another object in which the method specified by the method ID is executed, and

211

wherein said input data tag table generating means generates the input data tag table and adds the input data tag table to the method elements including the pointer to another object associated with the input data tag table.

26. An object-oriented program storage medium according to claim 25, wherein the first object having the method element to which the input data tag table is added calls, when calling the second object identified by the method element, the second object giving as arguments the method ID and the input data tag table which are stored in the method element.

27. An object-oriented program storage medium according to claim 26, wherein the second object receives messages directed from the first object to the second object, refers to the input data tag table, which is an argument of the received message, to obtain the pointer element list ID of the first object, produces the pointer element list identified by the pointer element list ID, of the first object and in addition the data element list identified by the data element list ID associated with the pointer element list ID, of the second, and writes the pointers arranged in the data element list of the second object into the pointer

storage areas indicated by the pointers arranged in the pointer element list of the first object.

28. An object-oriented program storage medium according to claim 26, wherein the second object has means for producing a third object, receives messages directed from the first object to the second object, refers to the input data tag table, which is an argument of the received message, to obtain the pointer element list ID of the first object, produces the pointer element list identified by the pointer element list ID, of the first object and in addition the data element list identified by the data element list ID associated with the pointer element list ID, of the third, and writes the pointers arranged in the data element list of the third object into the pointer storage areas indicated by the pointers arranged in the pointer element list of the first object.

29. An object-oriented program storage medium for storing a plurality of objects each having data and operations, said object-oriented program storage medium storing an object coupling program comprising:

instruction coupling means for permitting a transfer of messages between a first object having an output

213

instruction bus portion for performing a processing for an issue of messages directed to another object and a second object having an input instruction bus portion responsive to messages issued by another object and directed to self object for activating a method of self object associated with the received message, by means of providing such a correspondence that the message of the first object is associated with the method of the second object; and

an output data tag table generating means for generating an output data tag table indicating an association of a pointer element list ID for identifying a pointer element list in which pointers to pointer storage areas for storing pointers to data are arranged with a data element list ID for identifying a data element list in which pointers to data storage areas for storing data are arranged, for each other object, on the output instruction bus portion of self object.

30. An object-oriented program storage medium according to claim 29, wherein said instruction coupling means generates a method element list in which arranged are method elements including a method ID for specifying a method of another object associated with a message of self object, and a pointer to another object in which the method

specified by the method ID is executed, and

wherein said output data tag table generating means generates the output data tag table and adds the output data tag table to the method elements including the pointer to another object associated with the output data tag table.

31.   An object-oriented program storage medium according to claim 30, wherein the first object having the method element to which the output data tag table is added calls, when calling the second object identified by the method element, the second object giving as arguments the method ID and the output data tag table which are stored in the method element.

32.   An object-oriented program storage medium according to claim 31, wherein the second object receives messages directed from the first object to the second object, refers to the output data tag table, which is an argument of the received message, to obtain the data element list ID of the first object, produces the data element list identified by the data element list ID, of the first object and in addition the pointer element list identified by the pointer element list ID associated with the data element list ID, of the second, and writes the pointers arranged in the data

215

element list of the first object into the pointer storage areas indicated by the pointers arranged in the pointer element list of the second object.

33. An object-oriented program storage medium according to claim 31, wherein the second object has means for producing a third object, receives messages directed from the first object to the second object, refers to the output data tag table, which is an argument of the received message, to obtain the data element list ID of the first object, produces the data element list identified by the data element list ID, of the first object and in addition the pointer element list identified by the pointer element list ID associated with the data element list ID, of the third, and writes the pointers arranged in the data element list of the first object into the pointer storage areas indicated by the pointers arranged in the pointer element list of the third object.

34. An object-oriented program storage medium according to claim 12, wherein the second object receives messages directed from the first object to the second object, and executes the method identified by the method ID which is an argument of the received message.

216

35. An object-oriented program storage medium according to claim 15, wherein the second object receives messages directed from the first object to the second object, and executes the method identified by the method ID which is an argument of the received message.

36. An object-oriented program storage medium according to claim 21, wherein the second object receives messages directed from the first object to the second object, and executes the method identified by the method ID which is an argument of the received message.

37. An object-oriented program storage medium according to claim 26, wherein the second object receives messages directed from the first object to the second object, and executes the method identified by the method ID which is an argument of the received message.

38. An object-oriented program storage medium according to claim 31, wherein the second object receives messages directed from the first object to the second object, and executes the method identified by the method ID which is an argument of the received message.

39.   An object-between-network display method in which a plurality of objects produced by an object-oriented programming and wirings representative of flow of data and control among the plurality of objects are displayed on a display screen of an image display apparatus for displaying images based on electronic image information,

wherein displayed on the display screen is a first image in which a display area consisting of one measure obtained through partitioning the display screen into a plurality of measures, or a display area formed through coupling a plurality of adjacent measures together, comprises an object display domain for displaying a single object, and a wiring display domain for displaying wires for coupling a plurality of objects to one another, the object display domain and the wiring display domain are determined in such a manner that the wiring display domain is formed between the object display domain-to-object display domain of the adjacent two display areas, and

wherein on the display screen each of the plurality of objects is arranged on an associated object display domain of the display area, while the wires for coupling the plurality of objects thus arranged are displayed on the wiring display domains ranged across a plurality of display

218

areas.

40. An object-between-network display method according to claim 39, wherein a predetermined object of a plurality of objects constituting the first image is constituted of a subnetwork comprising a plurality of objects, which are of lower class in a hierarchical structure than the predetermined object, and wirings for connecting the later plurality of objects together, and

wherein when a second image, in which a subnetwork of said predetermined object is displayed instead of a display of said predetermined object in the first image, is displayed instead of the first image, the subnetwork on the first image is displayed in a more enlarged display area than that of said predetermined object, and display areas arranged upper and lower sides and right and left sides of the display area of the subnetwork are altered to display areas enlarged vertically and horizontally, respectively, and regarding display areas located at diagonal positions with respect to the display area of the subnetwork, the display areas are displayed with a same size as that of the first image.

41. An object-between-network display method according to claim 39, wherein a predetermined object of a

plurality of objects constituting the first image is

constituted of a subnetwork comprising a plurality of

objects, which are of lower class in a hierarchical structure

than the predetermined object, and wirings for connecting the

later plurality of objects together, and

wherein when a second image, in which a subnetwork of

said predetermined object is displayed instead of a display

of said predetermined object in the first image, is displayed

instead of the first image, the subnetwork on the first image

is displayed in a more enlarged display area than that of

said predetermined object, and display areas except the

display areas of the subnetwork are deformed as compared with

the associated display areas on the first image in such a

manner that display areas located at a periphery of the

second image, and position and size of sides contacting with

the second image, are substantially the same ones as display

areas located at a periphery of the first image, and position

and size of sides contacting with the first image,

respectively.

42.   An object-between-network display method

according to claim 39, wherein when the first image is

displayed, figures and sizes of the object display domains in

the display areas are standardized in accordance with figures

and sizes of the display areas.

43.   An object-between-network display method according to claim 39, wherein when the first image is displayed, first, the plurality of objects are displayed, and then it is displayed that the plurality of objects are interconnected with wirings in which a direction of flow of data or control is repeatedly displayed in units of predetermined segments.

44.   An object-between-network display method according to claim 39, wherein when the first image is displayed, in wirings consisting of a central wire and edge wires extended along both sides of the central wire, each of the edge wire having a display aspect different from the central wire, there is provided such a display of wiring that of the intersecting wirings, with respect to wirings each representative of a same flow of data or control, the central wire-to-central wire are continued, and with respect to wirings each representative of a mutually different flow of data or control, the central wire of one of the wirings is divided into parts at a position contacting with or adjacent to the edge wires of another wiring.

45. An object-oriented programming supporting apparatus for coupling a plurality of objects, each having data and operations, with one another in accordance with an instruction, said object-oriented programming supporting apparatus comprising:

display means for displaying objects each represented by a block representative of a main frame of an object, a data output terminal for transferring data of the object to another object, a data input terminal for receiving data from another object, a message terminal for issuing a message to make a request for processing to another object, and a method terminal for receiving a processing request from another object to execute a method, the object being represented by a hierarchical structure which permits one or a plurality of objects to exist in a single object, and in addition displays a wiring for coupling terminals of a plurality of objects;

object coupling means for constructing a coupling structure among a plurality of objects in accordance with an instruction for coupling terminals of the plurality of objects through a wiring;

hierarchical structure construction means for constructing a hierarchical structure of objects; and

a handler for instructing a wiring for coupling among objects to said object coupling means, and in addition for

222

instructing a position of an object on the hierarchical

structure to said hierarchical structure construction means,

wherein said hierarchical structure construction

means has means for producing a duplicate object of a

substantial object designated in accordance with an

instruction from said handler, and for disposing the

duplicate object at a hierarchy different from a hierarchy at

which the substantial object is disposed, and

said object coupling means receives from said handler

an instruction as to a wiring between the duplicate object

and another object in the wiring of the hierarchical

structure in which the duplicate object is disposed, and

constructs a coupling structure in which the duplicate object

and the associated substantial object are provided in the

form of a united object.

46. An object-oriented programming supporting

apparatus for coupling a plurality of objects, each having

data and operations, with one another in accordance with an

instruction, said object-oriented programming supporting

apparatus comprising:

display means for displaying objects each represented

by a block representative of a main frame of an object, a

data output terminal for transferring data of the object to

223

another object, a data input terminal for receiving data from another object, a message terminal for issuing a message to make a request for processing to another object, and a method terminal for receiving a processing request from another object to execute a method, the object being represented by a hierarchical structure which permits one or a plurality of objects to exist in a single object, and in addition displays a wiring for coupling terminals of a plurality of objects;

object coupling means for constructing a coupling structure among a plurality of objects in accordance with an instruction for coupling terminals of the plurality of objects through a wiring;

hierarchical structure construction means for constructing a hierarchical structure of objects; and

a handler for instructing a wiring for coupling among objects to said object coupling means, and in addition for instructing a position of an object on the hierarchical structure to said hierarchical structure construction means,

wherein said object coupling means releases a coupling structure of the object before a replacement with another object in accordance with an instruction from said handler, and causes the object after the replacement to succeed to the coupling structure of the object before the replacement with another object, and

224

said hierarchical structure construction means disposes the object after the replacement, instead of the object before the replacement, at a hierarchy at which the object before the replacement is disposed.

47. An object-oriented programming supporting apparatus for coupling a plurality of objects, each having data and operations, with one another in accordance with an instruction, said object-oriented programming supporting apparatus comprising:

display means for displaying objects each represented by a block representative of a main frame of an object, a data output terminal for transferring data of the object to another object, a data input terminal for receiving data from another object, a message terminal for issuing a message to make a request for processing to another object, and a method terminal for receiving a processing request from another object to execute a method, the object being represented by a hierarchical structure which permits one or a plurality of objects to exist in a single object, and in addition displays a wiring for coupling terminals of a plurality of objects;

object coupling means for constructing a coupling structure among a plurality of objects in accordance with an instruction for coupling terminals of the plurality of

225

objects through a wiring;

hierarchical structure construction means for constructing a hierarchical structure of objects; and

a handler for instructing a wiring for coupling among objects to said object coupling means, and in addition for instructing a position of an object on the hierarchical structure to said hierarchical structure construction means,

wherein said hierarchical structure construction means is in response to an instruction from said handler such that a plurality of objects from among the objects disposed at a predetermined hierarchy are designated and the plurality of objects are rearranged on the lower-order hierarchy by one stage, and rearranges the plurality of objects on the lower-order hierarchy by one stage, and produces and arranges an object including the plurality of objects on the predetermined hierarchy in such a manner that a coupling structure among the plurality of objects and a coupling structure among the plurality of objects and objects other than the plurality of objects are maintained.

48. An object-oriented programming supporting apparatus for coupling a plurality of objects, each having data and operations, with one another in accordance with an instruction, said object-oriented programming supporting

226

apparatus comprising:

display means for displaying objects each represented by a block representative of a main frame of an object, a data output terminal for transferring data of the object to another object, a data input terminal for receiving data from another object, a message terminal for issuing a message to make a request for processing to another object, and a method terminal for receiving a processing request from another object to execute a method, the object being represented by a hierarchical structure which permits one or a plurality of objects to exist in a single object, and in addition displays a wiring for coupling terminals of a plurality of objects;

object coupling means for constructing a coupling structure among a plurality of objects in accordance with an instruction for coupling terminals of the plurality of objects through a wiring;

hierarchical structure construction means for constructing a hierarchical structure of objects; and

a handler for instructing a wiring for coupling among objects to said object coupling means, and in addition for instructing a position of an object on the hierarchical structure to said hierarchical structure construction means,

wherein said display means has, in case of existence of a plurality of method terminals connected to one message

227

terminal designated in accordance with an instruction through said handler, means for displaying a list indicative of an execution sequence of a plurality of methods associated with the plurality of method terminals, and

said object coupling means has means for reconstructing a coupling structure in which the execution sequence of the plurality of methods appearing at the list displayed on said display means are altered in accordance with an instruction by said handler.

49. A program storage medium for use in an object-oriented programming, the program storage medium being adapted for storing therein a program to support an object-oriented programming for coupling a plurality of objects, each having data and operations, with one another,

wherein each of said objects is represented by a block representative of a main frame of an object, a data output terminal for transferring data of the object to another object, a data input terminal for receiving data from another object, a message terminal for issuing a message to make a request for processing to another object, and a method terminal for receiving a processing request from another object to execute a method, the object being represented by a hierarchical structure which permits one or

228

a plurality of objects to exist in a single object, and an instruction for coupling terminals of the plurality of objects through a wiring is given,

said program includes: object coupling means for constructing a coupling structure among a plurality of objects in accordance with the instruction for coupling terminals of the plurality of objects through a wiring; and hierarchical structure construction means for constructing a hierarchical structure of objects, and

said program storage medium stores such a program that said hierarchical structure construction means has means for producing a duplicate object of a substantial object designated in accordance with an instruction from said handler, and for disposing the duplicate object at a hierarchy different from a hierarchy at which the substantial object is disposed, and said object coupling means receives from said handler an instruction as to a wiring between the duplicate object and another object in the wiring of the hierarchical structure in which the duplicate object is disposed, and constructs a coupling structure in which the duplicate object and the associated substantial object are provided in the form of a united object.

229

50. A program storage medium for use in an object-oriented programming, the program storage medium being adapted for storing therein a program to support an object-oriented programming for coupling a plurality of objects, each having data and operations, with one another,

wherein each of said objects is represented by a block representative of a main frame of an object, a data output terminal for transferring data of the object to another object, a data input terminal for receiving data from another object, a message terminal for issuing a message to make a request for processing to another object, and a method terminal for receiving a processing request from another object to execute a method, the object being represented by a hierarchical structure which permits one or a plurality of objects to exist in a single object, and an instruction for coupling terminals of the plurality of objects through a wiring is given,

said program includes: object coupling means for constructing a coupling structure among a plurality of objects in accordance with the instruction for coupling terminals of the plurality of objects through a wiring; and hierarchical structure construction means for constructing a hierarchical structure of objects, and

said program storage medium stores such a program

230

that said object coupling means releases a coupling structure

of the object before a replacement with another object in

accordance with an instruction for the replacement of

objects, and causes the object after the replacement to

succeed to the coupling structure of the object before

the replacement with another object, and said hierarchical

structure construction means disposes the object after the

replacement, instead of the object before the replacement, at

a hierarchy at which the object before the replacement is

disposed.

51.   A program storage medium for use in an

object-oriented programming, the program storage medium being

adapted for storing therein a program to support an

object-oriented programming for coupling a plurality of

objects, each having data and operations, with one another,

wherein each of said objects is represented by a

block representative of a main frame of an object, a data

output terminal for transferring data of the object to

another object, a data input terminal for receiving data from

another object, a message terminal for issuing a message to

make a request for processing to another object, and a

method terminal for receiving a processing request from

another object to execute a method, the object being

231

represented by a hierarchical structure which permits one or a plurality of objects to exist in a single object, and an instruction for coupling terminals of the plurality of objects through a wiring is given,

said program includes: object coupling means for constructing a coupling structure among a plurality of objects in accordance with the instruction for coupling terminals of the plurality of objects through a wiring; and hierarchical structure construction means for constructing a hierarchical structure of objects, and

said program storage medium stores such a program that said hierarchical structure construction means is in response to an instruction such that a plurality of objects from among the objects disposed at a predetermined hierarchy are designated and the plurality of objects are rearranged on the lower-order hierarchy by one stage, and rearranges the plurality of objects on the lower-order hierarchy by one stage, and produces and arranges an object including the plurality of objects on the predetermined hierarchy in such a manner that a coupling structure among the plurality of objects and a coupling structure among the plurality of objects and objects other than the plurality of objects are maintained.

232

52. A program storage medium for use in an object-oriented programming, the program storage medium being adapted for storing therein a program to support an object-oriented programming for coupling a plurality of objects, each having data and operations, with one another,

wherein each of said objects is represented by a block representative of a main frame of an object, a data output terminal for transferring data of the object to another object, a data input terminal for receiving data from another object, a message terminal for issuing a message to make a request for processing to another object, and a method terminal for receiving a processing request from another object to execute a method, the object being represented by a hierarchical structure which permits one or a plurality of objects to exist in a single object, and an instruction for coupling terminals of the plurality of objects through a wiring is given,

said program includes: object coupling means for constructing a coupling structure among a plurality of objects in accordance with the instruction for coupling terminals of the plurality of objects through a wiring; and hierarchical structure construction means for constructing a hierarchical structure of objects, and

said program storage medium stores such a program

233

that said object coupling means has, in case of existence of a plurality of method terminals connected to one message terminal designated, means for making up a list indicative of an execution sequence of a plurality of methods associated with the plurality of method terminals, and means for reconstructing a coupling structure in which the execution sequence of the plurality of methods is altered in accordance with an alteration instruction of the execution sequence of the plurality of methods appearing at the list.

53.   A component storage medium for storing a component which serves as one object in combination with a predetermined existing software, said component including a method of issuing an event of the predetermined existing software through a firing by a message issued in other object.

54.   A component storage medium according to claim 53, wherein said component further includes a message for informing other object of that the event is issued through executing said method.

55.   A component storage medium for storing a component which serves as one object in combination with a predetermined existing software, said component including a

234

message for informing other object, upon receipt of

occurrence of a predetermined event of the predetermined

existing software, of that the predetermined event is

generated.

---

56.   A component builder apparatus comprising:

a first handler for selectively indicating making of

methods and messages;

a second handler for inputting an instruction of an

issue of a desired event of a predetermined existing

software; and

a component builder means for building a component

which serves as one object in combination with said existing

software, said component builder means serving, when making

of a method is instructed by an operation of said first

handler and a predetermined event of the existing software

is issued by an operation of said second handler, to make on

the component a method which fires with a message issued by

another object and issues the event, and serving, when

making of a message is instructed by an operation of said

first handler and an issue of a predetermined event of the

existing software is instructed by an operation of said

second handler, in response to an occurrence of the event,

to make on the component a message for informing other

235

objects of the fact that the event occurred.

57. An object-oriented programming supporting
apparatus comprising:

a component file for storing therein a component
which serves as one object in combination with a
predetermined existing software, said component including a
method of issuing an event of the predetermined existing
software through a firing by a message issued in other
object, and a message for informing other object of that the
event is issued through executing said method, and said
component being stored in said component file with respect to
one or more existing softwares;

a handler for inputting an instruction of an issue of
the event as to the existing software;

an event log file for storing a list for the events
as to one or more existing softwares, which are sequentially
issued in accordance with an operation of said handler; and

a component coupling means for taking out
sequentially the events from said event log file to combine a
message of a component including the message for informing
other object of that the same event as that taken out before
is issued and a method of a component including the method of
issuing the same event as that taken out now.

236